EVALUATION OF PROJECT SELECTION TECHNIQUES FOR PAVEMENT NETWORK MAINTENANCE AND REPAIR

A Scholarly Paper Presented to

The Faculty of The Geotechnical Engineering Program

University of Maryland at College Park

by

LT Thomas L. Wood, CEC, USN

In Partial Fulfillment

of the Requirements for the Degree of

Master of Science (Civil Engineering)

Approved:

Faculty Advisor/Date

Director, Geotechnical Program

ABSTRACT

Different approaches have been suggested for determining the optimal mix of repair projects for a pavement network. These methods range from random selection to sophisticated mathematical optimization models.

This paper presents an analysis of several questions regarding the effectiveness of three possible selection methods.

First, the performance of three separate single year project selection methods on different size networks is assessed over a broad funding spectrum. The results indicate that as funding levels increase, the benefit obtained by different selection methods converge. In addition, as the size of the network increases, the convergence tends to occur at progressively lower funding levels.

Second, the effect of the performance prediction models on these same selection methods is assessed by altering the coefficients of the models to predict both faster and slower deterioration of the network. The "select sets" of projects created by priority ranking selection and Knapsack IP selection at three separate funding levels are compared to determine how much variation is introduced by the changes in the performance prediction. With a 30% acceleration and deceleration of the deterioration curves, there was little change in the optimal project set created by either method.

Finally, a modified Monte Carlo model is used to assess the general shape of the solution space. The results suggest that the solution space is relatively flat except in the immediate vicinity of the optimum. This, in turn, suggests that a Monte Carlo approach to this problem would require a large number of trials to approximate the optimum. This finding conceptually supports findings in this study and others, as well as the intuitive observation, that random maintenance and repair strategies perform poorly compared to more rational approaches. Since only a few sets of repair projects are near the optimum, the chances of a random selection matching one of these near optimal project sets are relatively small.

INTRODUCTION

Providing the maximum benefit to the network for a given expenditure of Maintenance and Repair dollars is a primary goal of any pavement management system. It has been argued (Haas, et. al. 1985 & Lytton, 1985) that to be truly effective, a system must be able to:

1) select the best repair strategy for a given segment of the network
2) select that mix of repair projects for the entire network that provides the maximum benefit without exceeding the allowable budget

3) be able to consider the temporal distribution of projects

There are two distinct categories of optimization, Annual Optimization and True Optimization.

Annual optimization does not consider the temporal distribution of projects, instead it contains an implicit assumption that the combination of optimal project solutions for each given year will provide an optimal solution over time.

True optimization, however, answers these three questions simultaneously:

1) which repair strategy should be used for a given segment ?
2) which segments should be repaired ?
3) when should the repairs be accomplished ?

(Lytton, 1985).

In order to address these three questions simultaneously, a model or selection procedure must be complex and computationally intensive. In fact, the procedure to answer each question individually can range from simple to extremely complex. Is the additional computational effort involved in the more sophisticated approaches warranted? What effect do changing funding levels have on the effectiveness of different selection techniques? How sensitive are selection methods to the predictive accuracy of the performance models used to predict the conditions of the pavement? What is the shape of the solution space of this problem, and what effect might that shape have on the performance of selection methods?

This study was conducted as a single year analysis, and was limited to rehabilitation options involving AC Overlays only. Therefore, questions 1 and 3, the type and timing of projects are not considered. The focus is instead on answering the above questions in the context of the annual process of selecting segments to be repaired.

PREVIOUS WORK

Using the Illinois DOT Pavement Management System (ILLINET), Mohseni, et. al., 1993, studied the effectiveness of six different project selection methods in a multi-year context:

1) Total needs
2) Random selection
3) Ranking (worst-first)
4) Incremental Benefit Cost
5) Long Range Optimization
6) Linear Programming

Dr. Mohseni's study spanned a ten year analysis period, and attempted to maximize the benefits to the ted pavement network given a total budget for the network of $ 75 million, d uted evenly over the analysis period. A total needs analysis was performed first to determine all of the required repair projects for the selected road network, resulting in a total of $ 90.1 million in identified repair projects.

First, a random selection of projects to attempt to repair all segments below a critical performance threshold was done to provide a baseline criteria, and to show that purely random decisions performed poorly with respect to more rigorous approaches. This was followed by a project selection process using methods 3 through 6, above. The advant and disadvantages of each of the selection methods are outlined in Mohse . a 1993. Some of those points are highlighted here, along with additional observations:

1) A strictly "worst-first" ranking system is overly simplistic and is therefore not an especially practical system in real implementations. As will be discussed later, many of the "optimization" criteria that are implicitly contained in more developed ranking systems are left out of this approach. This could partially account for the extremely poor performance of the ranking approach in this study.

2) The random approach considered only those segments whose CRS value was less than 7 at the beginning year of the analysis period. All of these segments "qualified" for rehabilitation. A random number generator was then used to select the timing and the type of rehabilitation technique applied to the segment. In addition to being random, this approach has two distinct disadvantages. First, only those segments whose condition was poor enough at the beginning of the analysis period were considered for rehabilitation. This means that even as other segments fell below the threshold as the analysis progressed through time, they were ineligible for rehabilitation. Second, the random application of rehabilitation strategies to segments can result in poor benefit realization for each segment, since there is no guarantee that the strategy applied to the segment is even appropriate for the segment. A better approach might be to either use a reasonable "decision tree" process to determine the appropriate strategy, or pick only one repair option and then apply random selection of the segments and the timing. The latter approach is followed for the study outlined in this paper.

3) The long range optimization method cannot consider yearly budget constraints, only the overall budget for the analysis period. Hence, the practical utility of this method is limited, except as a comparison to other solutions.

4) The Benefit Cost Ratio and Incremental Benefit Cost Ratio techniques are annual optimization methods. In many ways they are conceptually similar to the single and multi-dimensional Knapsack Integer Programming techniques, respectively. Therefore, the results from the Knapsack technique will be compared to the results from the BC/IBC techniques.

5) The Linear Programming solution was used as a substitute for Integer Programming because of expected computational problems in solving the IP problem. While Dynamic Programming solutions of the IP problem become intractable rapidly with increasing numbers of decision variables, a number of other approaches are available. Many of these algorithms exhibit "reasonable" performance with "n" values into the tens of thousands (Martello and Toth, 1990). It is expected that an exact or approximate solution to even the complex multi-year problem is possible using some of these algorithms.

The funding level of $ 75 million used corresponds to approximately 83% of the total needs based on the needs analysis of $ 90.1 million. Therefore the results in Tables 1. and 2. will be used for comparison with the 80% funding level results in the conclusion of this study.

Using a benefit measurement of "Vehicle Miles Traveled on Acceptable Pavements" or VMT-A, the study showed that at a constant funding level methods 4-6 produced similar benefits, while methods 2 and 3 produced significantly lower benefits. The results are summarized in Table 1., below:

% Increase in VMT-A

Method	VMT-A, (billion vehicle miles)	% Increase
Random Selection	2.98	-
Worst-First Ranking	3.82	28%
IBC	5.64	89%
Long Range Optimization	6.02	102%
Linear Programming	5.63	89%

Table 1.

The VMT-A benefit criteria is a composite index, calculated as follows:

$$\text{VMT-A} = \text{AADT} * \text{Adequate Length} * \text{Life} * (365)$$

Where:

AADT: The Annual Average Daily Traffic
Adequate Length: Length of pavement above condition threshold
Life: Expected life

This benefit coefficient has the disadvantage of being strictly a "threshold" benefit. Any pavement which meets the minimum criteria to be called "adequate" is considered as part of the benefit for as long as it is expected to be adequate. No additional credit is taken for pavements that are better than adequate. This may be somewhat compensated for by the fact that presumably these segments would have a longer Life above the adequate threshold.

Another possible criteria for measuring the effectiveness of the repair strategies is the length weighted average condition index (in this case CRS). Using this criteria, the range of variation is considerably smaller, as shown in Table 2., below:

% Increase in CRS

Method	CRS-Years	% increase
Random Selection	6.49	-
Worst-First Ranking	6.74	4%
IBC	6.82	5%
Long Range Optimization	6.99	8%
Linear Programming	6.81	5%

Table 2.

This benefit is calculated as the difference of the areas under the CRS/Time curves for the segment. While this benefit coefficient takes credit for the full improvement to the pavement, it has the disadvantage of not considering the effects of traffic on network benefit. Since traffic level is not included in the benefits, a repair to a segment with a low AADT provides the same net benefit to the overall network as the same repair to a similar segment with a high AADT. In reality, this is not true, as a more heavily traveled segment benefits more travelers.

In addition to the above two criteria, Dr. Mohseni describes two additional criteria, "Added Pavement Life" and "User's Benefits." Added Pavement Life is similar to CRS above, but results in a benefit coefficient that does not account for either traffic levels or increases in serviceability above the threshold value, hence combining the disadvantages of the VMT-A and CRS criteria. User's Benefits attempts to calculate the cost of user operation over the original and rehabilitated pavement sections. The difference between these two values is the "benefit" to the user's from the rehabilitation. Given the difficulty of defining user's costs, this method may be difficult to use properly.

As an additional facet of his study, Dr. Mohseni varied the budget provided to the network as a percentage of the identified needs to study the effects on network performance using different project selection methods. For all four of the above benefit coefficients, the range in performance variation by

the different selection methods either remained constant or increased with increasing funding levels. As will be more fully described later, this contrasts with the results of this study.

ANALYSIS METHODOLOGY

General

The first phase of this study evaluated the effectiveness of three different project selection processes at various funding levels for a single year analysis. These processes are:

1) Random Selection
2) "Knapsack" Integer Programming
3) PMAP Priority Ranking

As in the Mohseni study, a method of random selection is used as a baseline criteria. However, the random selection is limited to the selection of the segments, not the type of rehabilitation option or the timing. As noted earlier, both the type of repair option and the timing are constrained by the scope of the study. Integer programming is used as an alternative to Incremental Benefit Cost (IBC). The Delaware DOT PMAP system's Priority Ranking is used instead of the "worst-first" ranking used by Dr. Mohseni. Multi-year analysis techniques were not considered.

Network to be Analyzed

The State of Delaware has developed, under contract with PCS/LAW Inc. a statewide pavement management system designated as PMAP (Smith, et. al. 1993). This system is composed of several modules, including:

1) Database
2) Analysis and Forecasting Modules
3) Report Generation

The analysis and forecasting module performs selection of projects to repair the network. The analysis takes place in two stages, answering sequentially optimization questions 1 and 2 noted above.

First, the pavement condition data contained in the database module is used to forecast the condition of the pavement, including the PSI, Surface Distress, Roughness Number and Friction, forward in time to the analysis year. These predicted values are used by the system to analyze the network and determine what projects, if any, need to be undertaken to improve the Safety, Serviceability, Friction, or Capacity of each segment. These are essentially "project level" decisions as defined by Cook and Lytton, 1987, except that again, timing is not considered. The analysis identifies all of the "now" needs and determines the appropriate repair to fix the problem "now."

Second, a priority is assigned to each segment based on the following factors:

1) functional class
2) AADT Level
3) "score level" defined as a composite condition index based on structural, surface distress, ride quality, friction, and overall pavement condition.

This resultant composite priority is then used to rank the projects in order. The values of the priority numbers for each of the above factors can be altered by the user to reflect changing network priorities. Budgetary constraints for each category of work, (safety, serviceability, etc.) are then applied, resulting in a select set of the most important projects that can be done with the available funding. This second stage analysis, essentially the "which projects" part of the "network level" decision process, is the focus of the remainder of this study.

PMAP allows the user to select roads by various criteria for inclusion in the analysis. Three separate road networks were chosen for the first stage of this analysis. The first network was relatively small, and was primarily used to develop the analysis process. Network 2 was a larger selection of segments, consisting solely of segments in the HPMS category. Network 3 was the final and largest set chosen and was selected in order to try to validate trends in the data apparent from the first two networks.

The list of required projects for each subset road network was developed using the needs analysis option in the PMAP program. This analysis develops an unconstrained list of projects required to maintain the network. In all cases, the default cost data contained in the PMAP program was used to develop the costs for each project identified by the needs analysis.

It was assumed for the purposes of this study that the decision tree used in PMAP to determine the type of repair to be performed produces an "optimal" repair strategy for the individual segments. However, since the options for repair have been limited by the scope of the study to AC Overlays, the decision process is essentially limited to thickness designs.

Objective Functions

There are two steps that must be accomplished for any rational project selection technique to be applied:

1) Quantify the costs and benefits of each proposed activity
2) Apply some reasonable decision process that selects a set of projects that provides the most benefit to the network as a whole without exceeding the allowable budget

Quantifying the benefits and costs is done by assigning "value" to each state of repair of the network and by determining the cost to achieve that state. The "reasonable" decision process can take any one of several forms, from priority ranking to optimization.

A significant difference between ranking methods and optimization methods is that optimization methods require the explicit enumeration of the objective functions and their associated costs and benefits (Lytton, 1985), while ranking systems contain these implicitly. As noted earlier, a simplistic ranking system ignores many criteria, while a more sophisticated system will tend to reflect the same objectives as those explicitly contained in an optimization objective function. For example, the benefit coefficient used in this analysis includes the AADT as a measure of the number of users who benefit from (or suffer from) changes in the condition of a pavement section. In the PMAP priority ranking system, three levels of AADT, (low, medium and high), are used to assign the composite priority. A distinct advantage to the optimization procedure is that the effects of each factor are readily apparent, where they may be somewhat hidden in a ranking system. The disadvantage is that the formulation of the objective functions can be difficult and time consuming.

The objective criteria for the benefit to be gained by improvements to a given section shou'd reflect the influence of the following factors:

1) Traffic levels: more heavily traveled segments contribute more to (and detract more from) the overall performance of the network.
2) Serviceability: segments with a higher serviceability are more beneficial to the network than segments with lower serviceability.

Therefore, a reasonable "benefit coefficient" is defined as follows:

$$b_j = \frac{AADT_j * l_j}{L} * \Delta PSI_j$$

Where:

b_j : benefit of repair of segment "j"
l_j: Length of segment "j"
L: Total length of network being analyzed
$AADT_j$: predicted AADT for segment "j"
Δ PSI_j: Increase in serviceability for segment "j"

In addition to the above two factors, in multi-year, multi-repair option analyses, the durability of repairs becomes important. Conversion of the above benefit coefficient to a multi-year, multi-option coefficient is accomplished by plotting the curve of the original benefit coefficient vs. time, and the benefit coefficient after repair vs. time, then determining the net area under the curves. The difference in areas of the benefit/time curve for the original segment and for

different repair options is the benefit gained by the network for each repair option. In the analysis conducted for this paper, the use of the area under the curves is not required due to the annual nature and the limitation to a single repair option. However, this approach would be useful in the multi-dimensional knapsack analysis discussed later.

This benefit coefficient is a synthesis of the length-weighted pavement condition (Average network CRS), and traffic (VMT-A) used by Dr. Mohseni. As a result, it avoids the limitations of the individual benefit coefficients noted above. Full credit is taken for the condition of the pavement above the minimum acceptable threshold, and traffic effects are accounted for. As noted above, the durability of the repairs is not explicitly included until the coefficient is extended over time. However, the greater the initial increase in serviceability (Δ PSI) from any repair, the longer the pavement can be expected to remain above a minimum serviceability.

This benefit coefficient is not expressed in dollars. The conversion of the above benefit coefficient to a "dollar benefit per point" is possible by following a process similar to that used by Dr. Mohseni in defining the User's Benefits category described earlier. However, it was not done for this study for two reasons:

1) The Integer Programming technique being used depends upon the relative ratios between the benefits and costs for each segment repair option. The introduction of a constant conversion factor into this term would not alter the relative position of each option.

2) The real dollar value of costs and benefits can be difficult to determine.

The costs associated with the decision to repair or not repair are as follows:

1) Cost of performing repair
2) Increased user costs during repair
3) Increased cost of repairs if repairs are deferred to a later time

Increased user costs during construction can be included in the project cost. However, as with other user costs they can be difficult to determine, and for this reason they are not included in this analysis. For purposes of a single year analysis, there is no increased cost to the network in deferring a project. Therefore, the cost coefficient reduces to:

c_j = (project cost)

The KNAPSACK Integer Programming Problem

Single Dimensional Problem

The optimization model used in this study is a classic example of the single dimensional "Knapsack Problem", in which a knapsack has a total maximum weight capacity, and a number of items can either be packed (decision variable =1) or not packed (decision variable = 0). This is summarized mathematically as follows:

Maximize $\sum_{j=1}^{n} b_j x_j$

Subject to $\sum_{j=1}^{n} c_j x_j \leq B$

and $x_j = 0$ or 1

Where:

n: number of segments in selected network
x_j: Decision variable, repair or not repair segment "j", constrained to be 1 or 0
B : Total repair budget not to be exceeded
c_j: Cost to repair segment "j"
and all others are as defined above

As with all optimization problems, any constraint which contains an inequality is converted to an equality by the addition of a "slack variable" of appropriate sign into the inequality. In this problem the budget constraint contains a " ≤ "constraint, therefore a slack variable is added to the budget constraint to convert it to an equality.

This problem is the simplest example of a Linear, Integer Programming problem and can be solved by any number of methods. A simple and straightforward method of solution is the Branch and Bound Algorithm (Salkin & Mathur, 1989). In addition, there are several heuristic approaches that are computationally efficient and provide satisfactory results. A heuristic approach was used here to simplify the computational process. Comparison of the solution obtained with this approach to the Upper Bound indicates that it provides a solution that is within 3 % of the Upper Bound.

The general solution approach to this problem is to calculate the benefit to cost ratio for each repair option, and then sort the options in descending order. The Linear Programming solution then becomes:

$x^* = (1,\ldots,1,b^*/c_t,0,\ldots,0)$

Where:

t is the smallest index which satisfies:

$\sum_{j=1}^{t} c_j \geq b$;

$b^* = b - \left[\sum_{j=1}^{t-1} x_j c_j\right]$;

and:

x^*: The maximum value of the objective function for the LP solution
b: The total budget not to be exceeded
c_j: Cost coefficient for segment "j"
x_j: Decision variable for segment "j", set to 1 for all segments where j=1 to t-1

Note that this solution contains one fractional decision variable, x_t, and is therefore an infeasible solution to the integer programming problem.

From this point there are a number of possible solutions to the IP problem. The Upper Bound is found by setting x_t to 1. This provides the maximum value of the objective function, but is an infeasible solution since it violates the budget constraint.

The Lower Bound is found by setting x_t equal to zero, thereby making the solution feasible, but producing a less than maximum value of the objective function.

By recognizing that the value of the slack variable in the budget constraint becomes greater than zero in the Lower Bound solution, the value of the objective function can be increased. Using a heuristic approach, additional projects are added to the list until the slack variable becomes zero, or until no further projects can be added without the slack variable becoming negative.

There are also a number of exact solution algorithms available for this type of problem (Balas & Zemel, 1984, Martello & Toth, 1990). Those methods that do not require the additional computational effort of sorting the options by benefit/cost ratio are typically more efficient for large problems.

It should be noted that this procedure is conceptually similar to the Benefit Cost Ratio and Incremental Benefit Cost Ratio (Shahin, et. al., 1985) methods.

Multi - Dimensional Problem

To solve the annual optimization problem for more than one available repair option, a series of "multiple-choice" constraints for each segment may be added. These constraints require only one repair option out of a set of possible options be applied to each segment. The true optimization problem requires the

additional inclusion of multiple yearly budget constraints. This results in the formulation of a "multi-period" knapsack problem. The combination problem of multiple choice constraints and the multi-period constraints is referred to as a "multi-dimensional knapsack problem." The general formulation of this problem is as follows (Salkin & Mathur, 1989):

$$\text{Maximize } \sum_{t=1}^{T}\left[\sum_{j=1}^{n} b_{tj}x_j\right]$$

$$\text{Subject to } \sum_{t=1}^{T}\left[\sum_{j=1}^{n} c_{tj}x_j \le B_t\right]$$

$$\text{and: } x_j = 0 \text{ or } 1$$

$$\sum_{j=1}^{p} x_j = 1 \quad (j = 1,\ 2, \ldots, p \text{ for each segment, for a total of "n" segments})$$

Where:

T : the total number of periods in the analysis
t : index for time periods
p : number of repair methods for each segment
B_t : repair budget not to be exceeded for period t
c_{tj}: Cost to repair segment "j" in period t, using method associated with that segment and repair method (member of the appropriate subset)
b_{tj}: Benefit from repairing segment "j" in period "t"

and all others are as defined above.

This multi-dimensional model, while complex, is still solvable using Branch and Bound Search Algorithms. The simpler, annual optimization model, with or without multiple choice constraints, can be solved for large values of "n" using a number of techniques (Martello and Toth, 1990). As noted earlier, dynamic programming solutions of the Knapsack IP problem are not efficient for large values of "n".

While this approach was not used in this analysis, it is a logical extension to the work outlined here. Expanding these results to a multi-dimensional analysis would allow more direct comparison with Dr. Mohseni's results, and will also provide a method of solution for the true optimization problem.

RESULTS

Effects of Funding Level

Each of the three networks was analyzed using all three of the project selection methods over a range of funding levels, based on a percentage of the total rehabilitation needs of the network. The funding level begins at the extreme low end of the range at 5 % of identifiable backlog, and continues through 90 % of identifiable backlog.

Comparison of Benefits Gained

The performance curves for each method, as a function of the funding level are provided below. The benefits gained from executing the project sets selected by each method for each network are plotted against the funding level.

Network 1:

1) Maintenance District:	North
2) Federal Aid Classes:	Primary Secondary
3) Functional Classes:	Other Expressway/Freeway Other Principal Arterial Minor Arterial
4) Surface Types:	PCC and AC

The resulting subset included 585 segments for a total of 61.06 miles of roadway. The needs analysis generated a list of rehabilitation projects for 425 of the 585 segments in the network. The total cost for all rehabilitation projects was $10,649,732.

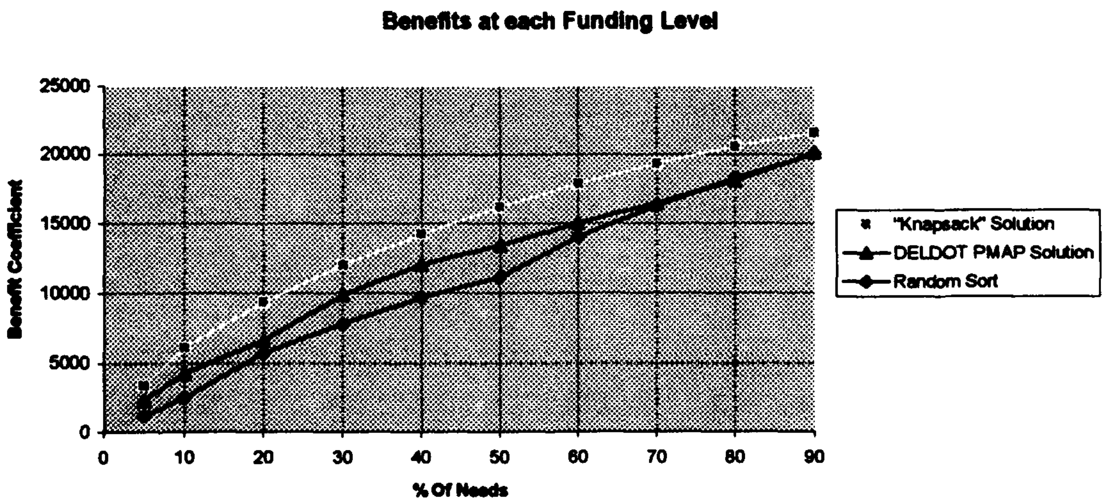

Figure 1.

The results from Network 1 can be summarized as follows:

1) The solutions for each method tend to converge at very both high and very low funding levels.

2) The largest absolute differential between the solutions occurs in the midrange funding levels, with decreasing differences as the solutions converge at either end of the funding spectrum.

3) The knapsack solution generally yielded an increase in benefits, on a percentage basis over the random solution, of twice the increase provided by the PMAP Priority solution.

4) At high funding levels, the benefits provided by the random solution approaches, and even exceeds those of the PMAP Priority solution, indicating that the priority ranking system picked a number of low benefit projects which drove the benefit coefficient down.

Network 2:

1) Section Type: HPMS Sections only
2) Roughness Number: .1<= SDI >=5
3) Surface Distress: .1<= SDI >=5HPMS Sections

The resulting subset included 708 segments for a total of 114.89 miles of roadway. This subset was limited to those segments included in the HPMS system since the data available for these segments tended to be more complete than that in the aggregate database. The needs analysis generated a list of rehabilitation projects for 509 of the 708 segments in the network. The total cost for all rehabilitation projects was $ 20,159,060.

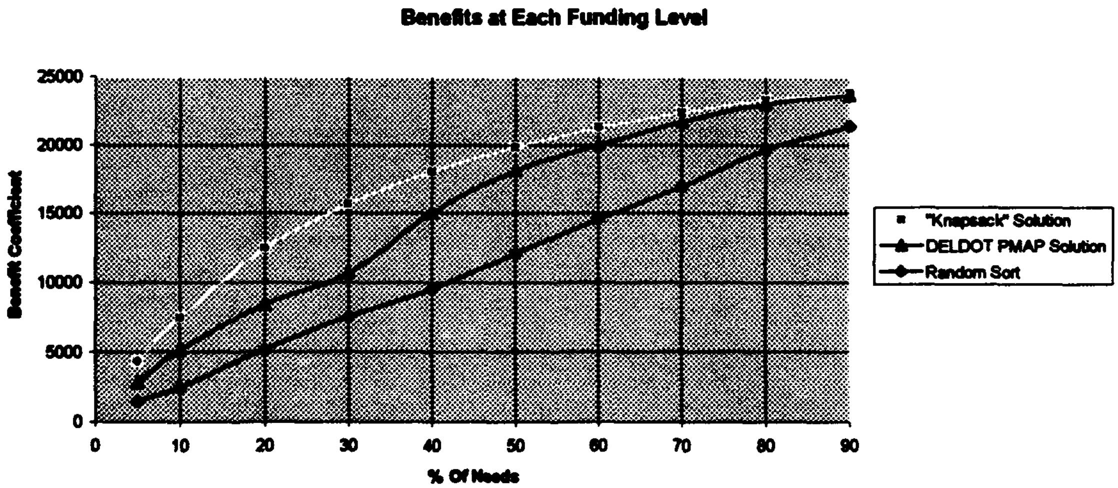

Figure 2.

The results from Network 2 can be summarized as follows:

1) As in the first network, the solutions for each method tend to converge at very both high and very low funding levels. However, the convergence is more pronounced.

2) The largest absolute differential between the solutions occurs in the lower midrange funding levels, shifted somewhat downward from the range observed in Network 1.

3) The difference between the Knapsack and PMAP Priority solutions was significantly smaller than in Network 1.

Network 3:

1) Maintenance District: North
2) Roughness Number: .1<= SDI >=5
3) Surface Distress: .1<= SDI >=5

The resulting subset included 3200 segments for a total of 352.19 miles of roadway. This large subset was deliberately limited to those segments that had a value other than "null" for the RN and SDI. This prevented the system from generating large numbers of project requirements for segments where the data was incomplete. The needs analysis generated a list of rehabilitation projects for 2245 of the 3200 segments in the network. The total cost for all rehabilitation projects was $75,183,724.

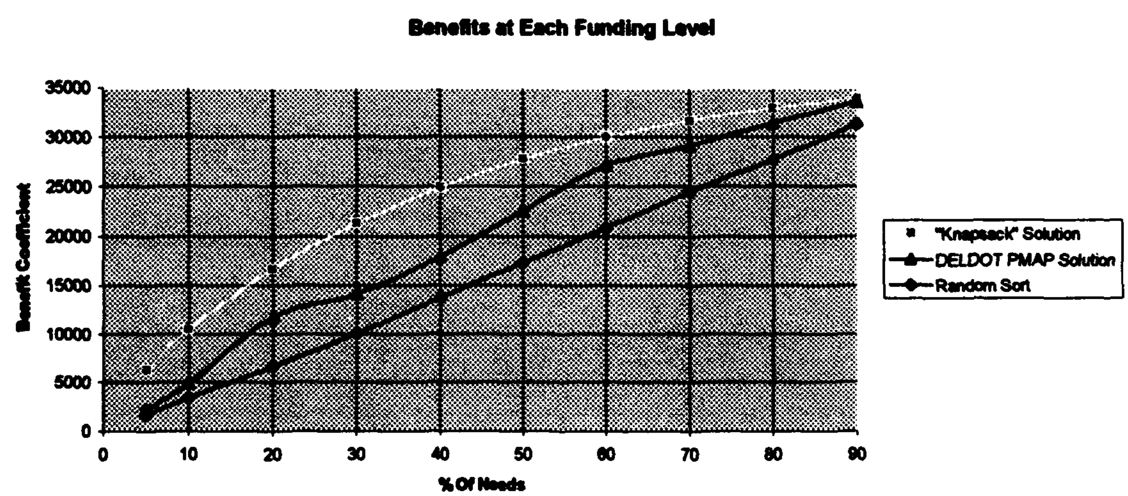

Figure 3.

The results from Network 3 can be summarized as follows:

1) The convergence between the random and PMAP Priority solutions is more pronounced at the higher end of the funding spectrum than in the Network 1, but somewhat less than that in Network 2. This may be an effect of the somewhat lower quality data in Network 3, (recall that Network 2 contained only HPMS sections), or there may in fact be no trend toward less unique solutions with increasing network size.

2) The largest absolute differential between the solutions occurs in the same lower midrange funding levels as in Network 2.

The overall performance of all three networks can be summarized by the following points:

1) As expected, the benefits accrued by all the networks increase with increasing funding.

2) With some minor variations in Network 1 which may be attributable to the relatively small size of the network, the shapes of the performance curves are markedly similar.

3) The Knapsack Selection method performed consistently better than the PMAP Priority Selection method, and both of these "rational" methods performed consistently better than the random selection method.

4) In Networks 2 and 3, the performance of PMAP Priority selection method improves relative to the Knapsack solution more quickly than the random selection as funding levels increase. At the 90% funding level, the priority selection approaches the value of the Knapsack solution. While all three solutions must arrive at the same repair strategy at 100% funding, the faster improvement of the priority selection suggests that more than just the inherent convergence of the solution may be responsible.

A sophisticated priority system such as PMAP's implicitly contains essentially the same objective criteria as the optimization routine contains explicitly. However, the overall performance of the PMAP Priority selection method was poorer than expected relative to the Knapsack solution. Specifically, the priority selection consistently performed poorly in the 20 to 40% funding ranges. In an attempt to determine the reasons for this relatively poor performance, an analysis of the repair strategies developed by each was conducted. This analysis uncovered a significant bias on the part of the PMAP system toward selecting long segments for repair.

PMAP Long Segment Bias

The PMAP priority ranking system shows a distinctive bias towards selecting longer segments for repair. This is especially marked at lower funding levels, see Figure 4. below. Figures 4 - 6 were developed from Network 1, however, similar results are observed in the other networks.

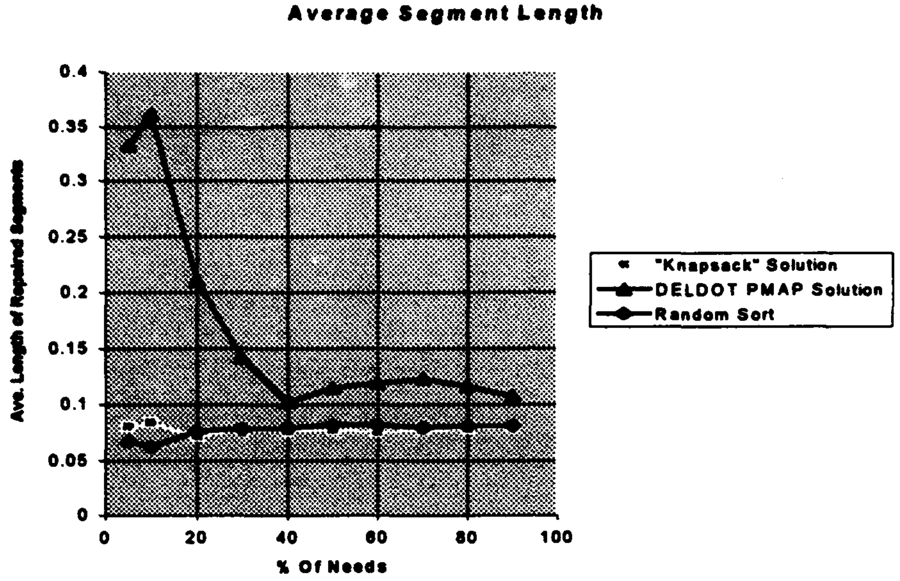

Figure 4.

The bias results in significantly fewer segments being repaired, as shown in Figure 5., although the total mileage of repaired road is not significantly different as shown in Figure 6.

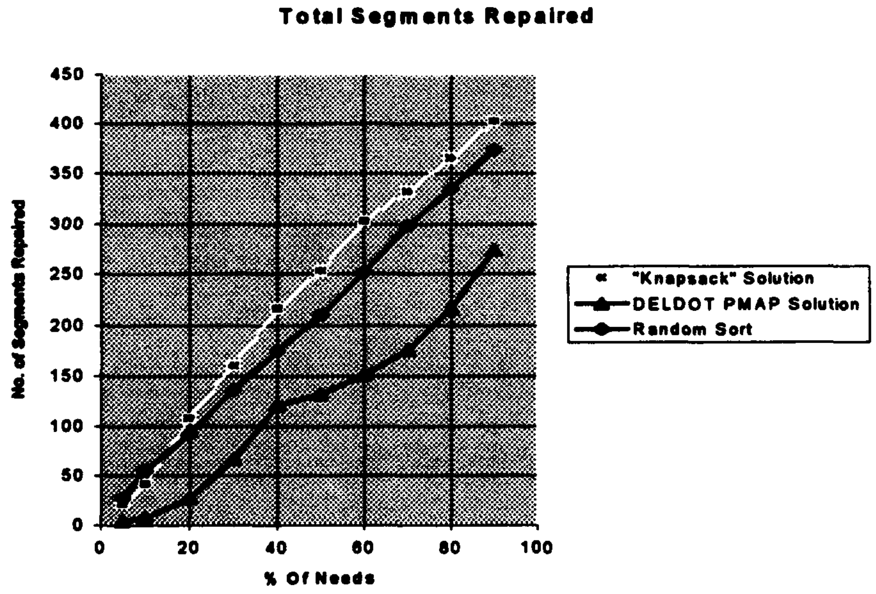

Figure 5.

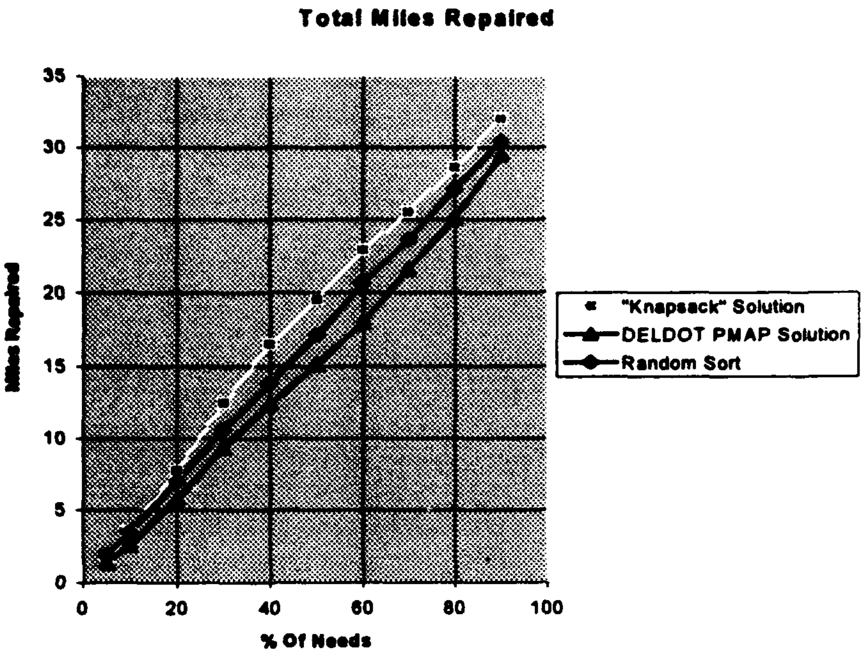

Figure 6.

This bias is the result of the secondary selection criteria used by the PMAP program to differentiate between segments of equal priority.

Since the PMAP priorities vary over a small range compared to the number of segments, many segments have equal priority according to their overall score level and the priority ranking factors. When this occurs a secondary criteria must be used to determine which of the projects will come first among each priority level. The PMAP program uses segment length. Those projects identified by the "needs analysis" are sorted based on priority, and then on segment length. A potential advantage of this criteria is to ensure that the longer segments, which are typically more expensive to repair, get funded. Without such a criteria, it is conceivable that at low funding levels large projects could remain in the backlog virtually forever.

Once the projects are sorted by the composite key of increasing priority number (lower priority segments have higher numbers) and descending length, a simple routine is used to select projects until funding is exhausted. Beginning at the top of the sorted project list, the search routine selects projects until the next project in the list would exceed the funding limit. At this point the routine "branches" and continues down the list until it finds a project it can afford. This project becomes the next "node" and is included in the project list. The search routine continues in a like fashion until either funding is totally exhausted or there are no projects that can be funded with the remaining funds.

Figure 4. indicates a decrease in the average segment length as the funding level increases, but it remains consistently higher than the other selection methods. At very low funding levels only the longest segment projects within the highest priority group are included, this results in a very long average segment length. Note that at the lowest funding levels, fewer of the longest segment projects can be afforded, hence a slightly lower average segment length is found. As funding levels increase, shorter projects enter into the solution, decreasing the average segment length. However, since as each new priority group enters the solution set, the segment length starts over at the longest segment in that priority group, the average will always be longer than any selection system that does not include a length bias.

In order to investigate the effects of different segment sorting methods inside each priority level, a simple trial using the 10% solution set from Network 2 was run. In this network priority level 4 was the highest priority identified, and at this funding level, only projects in priority level "4" were selected, as there was insufficient funding at 10% to fund any projects lower than priority level 4. In fact, not even all the priority level 4 projects were funded.

Three additional methods of sorting the priority level 4 projects were attempted. In each case a simple routine, similar to that used by the PMAP system, was used to select projects from the sorted project list until funds were exhausted. First, the projects were sorted by a random number key. Next, the list was sorted according to the predicted AADT over the segment. Finally, the projects were sorted by B/C ratio. The results are plotted in Figure 7. The B/C ratio sort gave the best benefit, however, the amount of increase was small compared to the "optimal" solution given by the Knapsack solution.

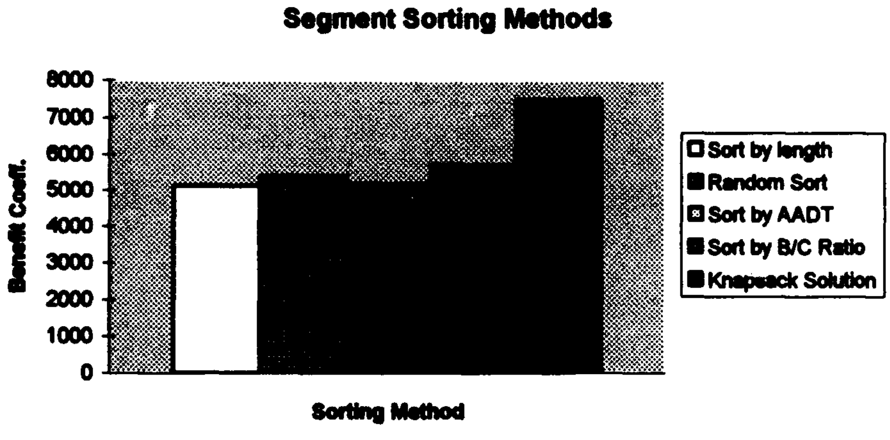

Figure 7.

This would tend to indicate that the secondary sorting criteria has a small effect on the overall performance of the ranking system. Therefore the most benefit will be achieved by improvements to the primary ranking criteria, thereby getting more of the best projects into the higher priority categories. However, the sorting criteria of B/C ratio would be fairly simple to implement in an existing system and may be warranted given its slightly better performance and somewhat more intuitively satisfying nature.

Effects of Changes in Performance Parameters

In the second phase of this work, the performance models used by the PMAP program to predict the condition of a pavement segment were altered to determine if such alteration significantly altered the projects selected for the optimal solution.

PMAP uses Surface Distress Index (SDI) and Roughness Number (RN) to calculate the value for Pavement Serviceability Index (PSI). Therefore, the coefficients of the RN and SDI performance models were altered in such a way that the pavement would deteriorate 30 % faster and 30 % slower than the "normal" or default models. Note that this does not mean that the actual values of RN or SDI were 30 % higher or lower than those given by the default models. For reasons that will be described later, the network selected contained only segments with SDI and RN values greater than 2. As a result the segments were on the relatively flat portion of the deterioration curve, even in the accelerated model. Therefore, the difference in predicted PSI was considerably less than 30 %, (5 to 10 %) in most cases.

The performance models used to predict the RN and SDI values in PMAP take the form of cubic polynomials as shown below:

$$RN = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

$$SDI = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$

By accelerating and decelerating the time base for each equation, modified performance curves for each model were created. Figures 7. and 8., below illustrate the modified performance curves.

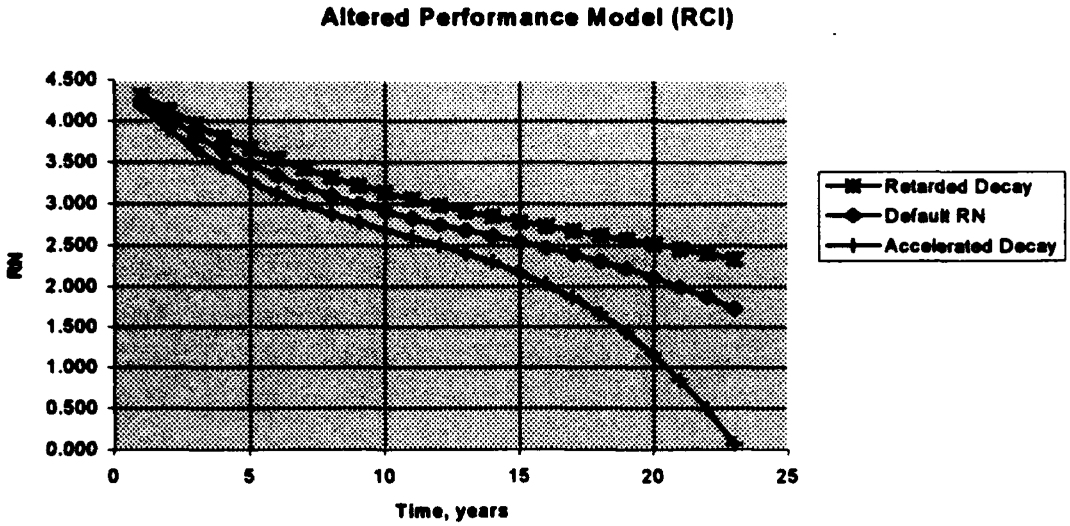

Figure 7.

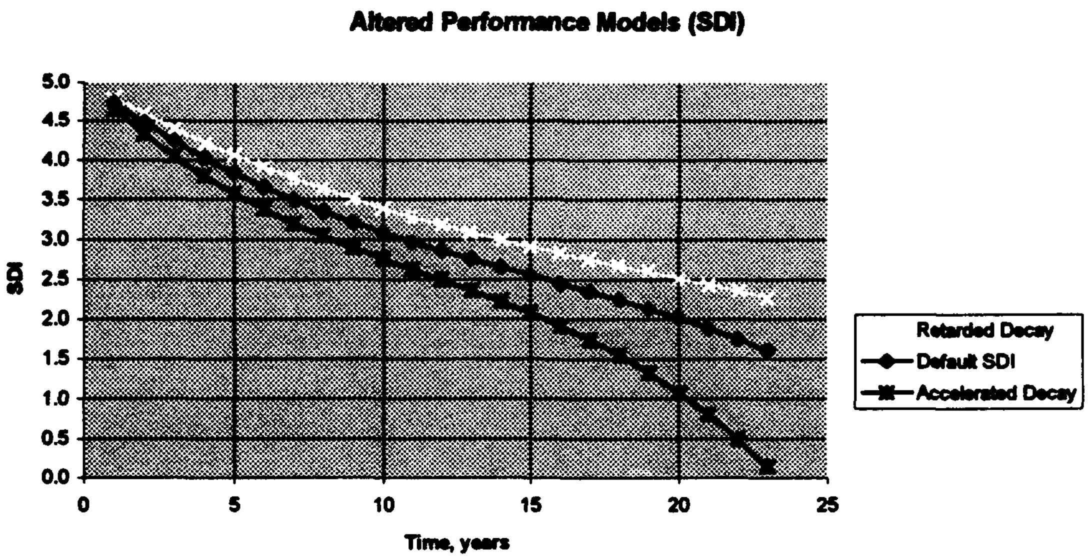

Figure 8.

The modified coefficients used to generate Figures 7. and 8. are listed in Tables 3. and 4., below.

RN Model Coefficients

	Defaults	Ret. Decay	Acc. Decay
a0	4.5105000	4.5105000	4.5105000
a1	-0.2709300	-0.2084000	-0.3522000
a2	0.0145500	0.0086000	0.0248000
a3	-0.0003500	-0.0001590	-0.0007693

Table 3.

SDI Model Coefficients

	Defaults	Ret. Decay	Acc. Decay
a0	5.0107000	5.0107000	5.0107000
a1	-0.2911900	-0.2240000	-0.3785000
a2	0.0126600	0.0075000	0.0214000
a3	-0.0002800	-0.0001280	-0.0006150

Table 4.

These modified model parameters were applied to a fourth network, selected as follows:

Network 4:

1) Maintenance District: All
2) Pavement Type: Flexible
3) Roughness Number: 2 <= SDI >= 5
4) Surface Distress: 2 <= SDI >= 5

The resulting subset included 786 segments for a total of 119.92 miles of roadway. This network was selected for two reasons. First, by choosing only one pavement type, the recalibration of the model parameters only needed to be done once. Second, by selecting RN and SDI values greater than 2, segments with no data and the very poor segments that could have predicted RN and SDI values of 0.0 using the default models were eliminated. This was intended to prevent a bias in the results from the majority of the segments needing repair regardless of the performance model.

Since many of the pavement sections included in the selection had RN and SDI dates in 1992 and 1993, 1996 was used as the analysis year for this procedure, so that the effects of the more or less rapid deterioration would be more distinct. Using the default performance model parameters, the needs analysis generated a list of rehabilitation projects for 440 of the 786 segments in the network. The total cost for all of these rehabilitation projects was $11,341,977. After altering the performance model parameters to allow for both faster and slower deterioration, similar needs analyses were run, also for analysis year 1996. A compilation of the results is shown below:

Total Projects Identified by Default Model : 440

Total Projects Identified by Accelerated Decay Model: 456

Total Projects Identified by Retarded Decay Model: 419

Using the PMAP Priority and Knapsack solutions, project select sets were selected at three funding levels. Budget levels of 5%, 30% and 60% of the total needs identified by the default performance model needs analysis were used.

By comparing the segments selected for repair by each method, a "% Concurrence" factor was determined. This factor was defined as follows:

$$\% \text{ Concurrence} = \frac{\text{Number of Projects Consistent w / Default Model}}{\text{Total Number of Projects Selected}} \bullet 100$$

Tables 5. through 10. list the results for each selection procedure at each funding level.

5% Funding, PMAP Priority Selection

Performance Model	Total Projects Selected	Number Consistent w/ Defaults	% Concurrence
Default	20	N/A	N/A
Retarded Decay	17	15	88%
Acc. Decay	25	18	72%

Table 5.

30% Funding, PMAP Priority Selection

Performance Model	Total Projects Selected	Number Consistent w/ Defaults	% Concurrence
Default	104	N/A	N/A
Retarded Decay	101	94	93%
Acc. Decay	108	101	94%

Table 6.

60% Funding, PMAP Priority Selection

Performance Model	Total Projects Selected	Number Consistent w/ Defaults	% Concurrence
Default	254	N/A	N/A
Retarded Decay	241	241	100%
Acc. Decay	260	250	96%

Table 7.

5% Funding, Knapsack Selection

Performance Model	Total Projects Selected	Number Consistent w/ Defaults	% Concurrence
Default	35	N/A	N/A
Retarded Decay	33	30	91%
Acc. Decay	38	33	87%

Table 8.

30% Funding, Knapsack Selection

Performance Model	Total Projects Selected	Number Consistent w/ Defaults	% Concurrence
Default	165	N/A	N/A
Retarded Decay	141	141	100%
Acc. Decay	157	147	94%

Table 9.

60% Funding, Knapsack Selection

Performance Model	Total Projects Selected	Number Consistent w/ Defaults	% Concurrence
Default	310	N/A	N/A
Retarded Decay	277	277	100%
Acc. Decay	304	296	97%

Table 10.

The above results indicate that accelerating or retarding the deterioration of a pavement, within a certain range, has little or no impact on the selected projects. In addition, the following points should be noted:

1) The PMAP long segment bias is responsible for the lower number of segments selected by PMAP.
2) The trend for both selection systems is increasing concurrence as funding levels increase.
3) The PMAP system has lower concurrence at any given funding level than the knapsack, however, that may be the result of the fewer segments being selected by the PMAP system.
4) The concurrence of the accelerated model is slightly less than that of the retarded model. This is because in the accelerated case, the pavements are on a steeper gradient of the curve. This indicates that at some point the difference in calculated PSI does start to make a difference in the projects selected.

Shape of Solution Space

The final phase of this study attempts to assess the shape of the solution space for this problem. Is the solution space highly curved, or relatively flat? In order to investigate this, the random project selection method was modified to perform multiple selections at each funding level. A series of trials were executed, each resulting in a new solution set with new benefit coefficients. By using a program written in Microsoft VISUAL BASIC, (see Appendix B for partial program listing), 1000 random project selections were made to meet each funding level. The results were then sorted in increasing order for each funding level. Therefore trial 1 is the lowest value of 1000 trials at each funding level, regardless of which trial actually produced that benefit level. The readability of the figure was improved by arranging the trials in order of increasing benefit, this

does not imply that trials themselves produced increasing benefits as the trial number increased. Also in the interests of readability, the results were averaged, 100 trials at a time to produce a total of 10 data points. These 10 data points, along with the highest and lowest randomly generated values, and the Knapsack solution benefit coefficient for the same set of road segments were then plotted in Figure 9. The Knapsack solution is the highest value in all cases and is plotted at the far right of Figure 9.

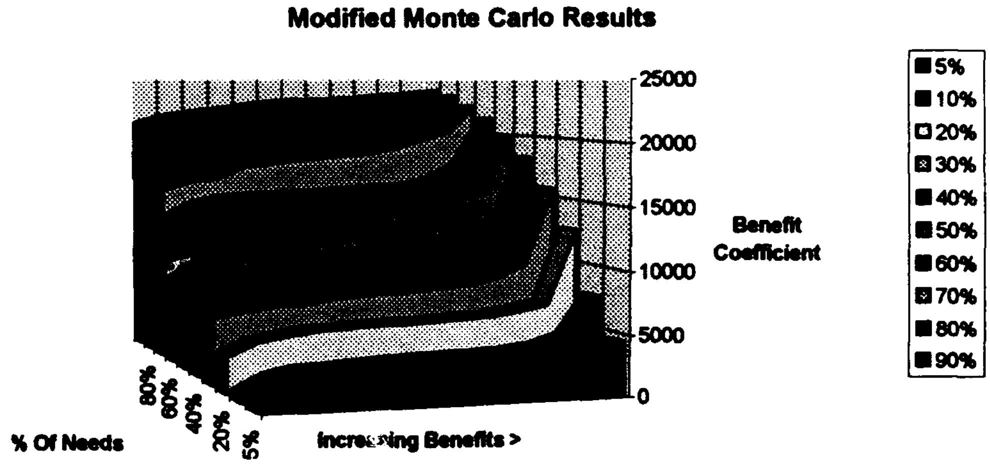

Figure 9.

While the number of trials executed is small compared to a what would be required for a true Monte Carlo optimization routine, the general shape of Figure 13 indicates that the majority of the feasible solutions to this problem fall within a narrow benefits band, and that the solution space is fairly flat, with a steeper gradient immediately around the optimum. This would suggest that Monte Carlo routines would be an especially inefficient choice for solution of this problem. However it should be noted that the difference between the "optimal" solution and the highest random selection decreases at very high funding levels, which is expected, since as noted earlier, the solutions converge at very high funding levels.

OPPORTUNITIES FOR FURTHER RESEARCH

Further Effects of Network Size

The trend in the networks selected indicates a tendency for less unique solutions as network size increases, as indicated by the somewhat earlier convergence of the performance curves. This may indicate that the solution space becomes flatter with increasing network size as well as increasing funding. Additional research could be done on the effects of network size on the uniqueness of the optimal repair strategy.

Multi-Dimensional Knapsack Solutions

The logical extension of the first phase of this work would be the inclusion of multiple choice constraints to allow for selection of multiple repair options for each segment. Following this stage, the extension of the work to a multi-period analysis would result in a "true" optimization model.

Dr. Mohseni states in his 1993 paper that Integer Programming solutions for this problem were not possible. He therefore used Linear Programming, which, because of the integer nature of the problem, does not guarantee an optimal solution. Multi-dimensional Knapsack solutions to both the annual and true optimization problems may provide an approach that will provide either an exact or approximate integer solution to the problem.

Sorting Methods for Segments of Equal Priority

Since any practical priority ranking system will always have a small number of priority levels compared to the number of segments being ranked, this "secondary sorting criteria" is of some importance. Even though the results of this study show that altering the secondary sorting criteria did not raise the performance of the priority ranking to the level of the Knapsack solution, there was still an increase of almost 11% in benefits by switching from the length based criteria to the B/C ratio criteria. Depending on the costs of implementation, it may be worthwhile to examine using different secondary sorting criteria to enhance the performance of existing priority ranking systems. At the least, designers of new priority ranking implementations should give careful consideration to the secondary criteria used since it does appear to have an impact on performance.

Effects of Altering Priority Factors

Altering the priority factors used to determine the relative priority of each segment in the network may have a significant affect on the solution provided by the priority ranking system. Since altering the secondary sorting criteria resulted in only a small increase in performance, it is reasonable to assume that any additional benefit gained from this system would have to be from altering the primary sorting criteria, which are the priority factors.

Further Effects of Performance Models

Altering the performance of a pavement over time has some effect on the optimal repair strategy. However, the results of this study indicate that the effect may not be as pronounced as previously believed. Further research into this question could focus on the effects over a broad spectrum of model performance, thus providing a "performance curve" that would indicate the range over which model performance has minimal effect on the optimal repair strategy

selected by a given method. These curves would probably be unique for each selection method. They could then be used to determine the level of accuracy needed in a performance model to avoid adversely affecting the optimal repair strategy selection process used in a particular pavement management system.

Differences Between Optimal Annual and True Solutions

Following extension of the annual knapsack solution presented here to a true optimization model, a comparison between the projects selected for execution under each could be done. The degree of concurrence between the projects selected for execution by each approach would be indicative of whether the additional computational effort involved in the multi-dimensional solution is worthwhile.

CONCLUSIONS

Phase 1

The results of the Mohseni study indicated a significant increase in the overall performance of the network by using any of the optimization techniques over the ranking system and random selection. This basic premise is borne out in this study across a broad spectrum of funding levels. The performance of any rational project selection procedure is better than a random or ad hoc approach; and optimization models perform better than even a relatively sophisticated priority ranking system. Table 11. lists the percent increase in benefits obtained by using both the PMAP Priority and the Knapsack IP solutions over the Random approach for Network 3.

% Increase in Benefits @ 80% Funding

Funding Level	PMAP % Increase over Random	Knapsack % Increase over Random
5	33%	274%
10	43%	210%
20	78%	153%
30	41%	112%
40	30%	80%
50	30%	60%
60	30%	44%
70	19%	29%
80	13%	19%
90	7%	8%

Table 11.

For ease of reference, Table 1. is reproduced below:

% Increase in VMT-A

Method	VMT-A, (billion vehicle miles)	% Increase
Random Selection	2.98	-
Worst-First Ranking	3.82	28%
IBC	5.64	89%
Long Range Optimization	6.02	102%
Linear Programming	5.63	89%

Table 1.

Comparing Tables 1. and 11. shows that at the 80% funding level, performance variation between the Randc.n Selection, Worst-First Ranking, and IBC Optimization methods varies over a considerably larger range in the Mohseni study than was found between the Random, PMAP Priority and Knapsack solutions used here. In addition, the convergence of the solutions at the extremes of the funding spectrum found in this study are not apparent in Dr. Mohseni's work.

Three additional points should be highlighted about these results:

1) The performance of the ranking system is especially poor at lower midrange funding levels, and steadily improves with increasing funding. Therefore, depending on the level of funding available, the priority system may provide adequate solutions.

2) No attempt was made to alter the priority coefficients used in the PMAP program from their default values. It is possible that "tweaking" these factors could significantly improve the performance of the system.

3) The heuristic Knapsack solution applied to this problem is <u>extremely</u> simple and easy to implement. An exact branch and bound solution is not much more difficult. Given the simplicity of these methods, they may warrant implementation in place of existing priority ranking systems.

Phase 2

The affects of altering the performance models were found to be considerably less than currently believed. A modest acceleration and deceleration of the pavement deterioration had virtually no effect on the project sets selected for execution at reasonable funding levels. This was true for both the PMAP and Knapsack solutions, even though the two selection methods provided different sets of projects and different net benefits to the network.

Phase 3

Based on the results of the modified Monte Carlo routine, the shape of the solution space of this problem is estimated to be relatively flat, except in the immediate vicinity of the optimum. The immediate significance of this finding is that the optimal repair strategy is fairly unique. This means that small changes in the projects selected for execution can have a significant impact on the overall performance of the repair strategy, as measured by the benefit gained. However, as the funding available to repair the network increases as a percentage of the total needs, the "uniqueness" of the optimal solution decreases. This is illustrated by both Figure 13., and by the convergence of the performance curves (Figures 1., 2., and 3.) at the higher funding levels.

Appendix A. References

Balas, E., and E. Zemel, (1980). "An Algorithm for Large Zero-One Knapsack Problems," *Operations Research*, Vol. 28, p.p. 1130-1154

Bunday, Brian D. (1984). Basic Linear Programming. Edward Arnold, (Publishers), Ltd. London, England

Cook, Wade D., and Robert L. Lytton, (1987). "Recent Developments and Potential Future Directions in Ranking and Optimization Procedures for Pavement Management" Presented at the 2nd North American Conference on Pavement Management

Dyer, M. E., (1984). "An 0(n) Algorithm for the Multiple Choice Knapsack Problem." *Mathematical Programming*, Vol. 29, No. 1, p.p. 57-63

Haas, Ralph, M. A. Karan, A. Cheetham, and Sue Khalil, (1985). "Pavement Rehabilitation Programming: A Range of Options" Presented at the 1st North American Conference on Pavement Management

Jewell, Thomas K. (1986). A Systems Approach to Civil Engineering Planning and Design. Harper & Row, Publishers, Inc. New York, New York

Lytton, Robert L. (1985). "Moderator Report" Presented at the 1st North American Conference on Pavement Management

Markow, Michael J., Brian D. Brademeyer, James Sherwood, and William J. Kenis, (1987). "The Economic Optimization of Pavement Maintenance and Rehabilitation Policy" Presented at the 2nd North American Conference on Pavement Management

Martello, Silvano, and Paolo Toth, (1990). Knapsack Problems: Algorithms and Computer Implementations. John Wiley & Sons, Sussex, England.

Mohseni, A., Michael I. Darter, and James P. Hall, (1991). "Effect of Selecting Different Rehabilitation Alternatives and Timing on Network Performance" Pavement Management Implementation, ASTM STP 1121, Frank B. Holt, Wade L. Gramling, Editors, American Society for Testing and Materials, Philadelphia, Pennsylvania

Pavement Management System for the State of Delaware, Project Report - Phase II, Volume II - PMAP User's Manual. (1992). PCS/LAW Engineering, Beltsville, Maryland

Salkin, Harvey M. and Kamlesh Mathur, (1989). Foundations of Integer Programming. Elsevier Science Publishing Co., Inc. New York, New York

Shahin, M. Y., S. D. Kohn, R. L. Lytton, and W. F. McFarland, (1985). "Pavement M&R Budget Optimization Using the Incremental Benefit-Cost Technique" Presented at the 1st North American Conference on Pavement Management

Smith, Harry A., Gonzalo R. Rada, Peter J. Stephanos, Charles W. Schwartz, and Lorenzo Casanova, (1993). "Development of Delaware DOT Pavement Management System" Presented at the 72nd Annual Meeting of the Transportation Research Board

Appendix B. Monte Carlo Program BASIC Code

The code listing below is the "meat" of the Monte Carlo routine written to perform the analysis of the shape of the solution space. The various forms with their associated controls that are part of the Visual Basic environment are not included.

```
Sub Command1_Click ()
Rem Set variables
Rem Can be set by VB Controls if program is to be
Rem used for multiple trials on different data files
numseg = 509
t = 1000
budget = 20159060
Rem dimension arrays
ReDim ax(10, numseg), temp(10), results(20, t), temp2(2)
Rem read data from file
Open "c:\vb\samples\data1.csv" For Input As #1
For x = 1 To numseg
For y = 2 To 10
Input #1, ax(y, x)
Next y
Next x
Rem build loop for multiple trials
For count1 = 1 To t
Rem add random sort key to array
For y = 1 To numseg
Randomize Timer
ax(1, y) = Rnd(12398)
Next y
Rem Sort routine for putting segments in random key order
n1 = numseg - 1
For x = 1 To n1
y = x
z = y + 1
For i = z To numseg
If ax(1, y) > ax(1, i) Then y = i
Next i
For i = 1 To 8
temp(i) = ax(i, x)
ax(i, x) = ax(i, y)
ax(i, y) = temp(i)
Next i
Next x
Rem Add up benefits for each funding level
totalben = 0
totalcost = 0
For count2 = 1 To numseg
totalben = totalben + ax(10, count2)
totalcost = totalcost + ax(7, count2)
If Abs(totalcost - .05 * budget) < Abs(.05 * budget - results(1, count1)) And (totalcost - .05 *
budget) <= 0 Then results(1, count1) = totalcost: results(2, count1) = totalben
```

```
If Abs(totalcost - .1 * budget) < Abs(.1 * budget - results(3, count1)) And (totalcost - .1 * budget)
<= 0 Then results(3, count1) = totalcost: results(4, count1) = totalben
If Abs(totalcost - .2 * budget) < Abs(.2 * budget - results(5, count1)) And (totalcost - .2 * budget)
<= 0 Then results(5, count1) = totalcost: results(6, count1) = totalben
If Abs(totalcost - .3 * budget) < Abs(.3 * budget - results(7, count1)) And (totalcost - .3 * budget)
<= 0 Then results(7, count1) = totalcost: results(8, count1) = totalben
If Abs(totalcost - .4 * budget) < Abs(.4 * budget - results(9, count1)) And (totalcost - .4 * budget)
<= 0 Then results(9, count1) = totalcost: results(10, count1) = totalben
If Abs(totalcost - .5 * budget) < Abs(.5 * budget - results(11, count1)) And (totalcost - .5 * budget)
<= 0 Then results(11, count1) = totalcost: results(12, count1) = totalben
If Abs(totalcost - .6 * budget) < Abs(.6 * budget - results(13, count1)) And (totalcost - .6 * budget)
<= 0 Then results(13, count1) = totalcost: results(14, count1) = totalben
If Abs(totalcost - .7 * budget) < Abs(.7 * budget - results(15, count1)) And (totalcost - .7 * budget)
<= 0 Then results(15, count1) = totalcost: results(16, count1) = totalben
If Abs(totalcost - .8 * budget) < Abs(.8 * budget - results(17, count1)) And (totalcost - .8 * budget)
<= 0 Then results(17, count1) = totalcost: results(18, count1) = totalben
If Abs(totalcost - .9 * budget) < Abs(.9 * budget - results(19, count1)) And (totalcost - .9 * budget)
<= 0 Then results(19, count1) = totalcost: results(20, count1) = totalben
Next count2
Next count1
Rem Sort results in increasing benfit order
For count6 = 2 To 20 Step 2
n2 = t - 1
For x = 1 To n2
y = x
z = y + 1
For i = z To t
If results(count6, y) > results(count6, i) Then y = i
Next i
temp2(1) = results(count6, x)
results(count6, x) = results(count6, y)
results(count6, y) = temp2(1)
temp2(2) = results(count6 - 1, x)
results(count6 - 1, x) = results(count6 - 1, y)
results(count6 - 1, y) = temp2(2)
Next x
Next count6
Open "c:\vb\samples\result.out" For Output As #2
For count3 = 1 To t
Write #2, results(1, count3), results(2, count3), results(3, count3), results(4, count3), results(5,
count3), results(6, count3), results(7, count3), results(8, count3), results(9, count3), results(10,
count3), results(11, count3), results(12, count3), results(13, count3), results(14, count3),
results(15, count3), results(16, count3), results(17, count3), results(18, count3), results(19,
count3), results(20, count3)
Next count3
Close #2
End Sub
```